A rational model of function learning

A rational model of function learning

Christopher G. Lucas[1], Thomas L. Griffiths[2], Joseph J. Williams[3], Michael L. Kalish[4]

1. School of Informatics, University of Edinburgh.

2. Department of Psychology, University of California, Berkeley.

3. HarvardX, Harvard University.

4. Department of Psychology, Syracuse University.

Address for correspondence:
Christopher G. Lucas
University of Edinburgh
School of Informatics
10 Crichton St.
Edinburgh, EH8 9AB
E-mail: c.lucas@ed.ac.uk

Author Note

This work was supported by AFOSR grant number FA9550-07-1-0351, NSERC and SSHRC Canada, and the McDonnell Causal Learning Collaborative. Preliminary results from the first two simulations were presented at the Neural Information Processing Systems conference (Griffiths, Lucas, Williams, & Kalish, 2009).

Abstract

Theories of how people learn relationships between continuous variables have tended to focus on two possibilities: that people are estimating explicit functions, or that they are performing associative learning supported by similarity. We provide a rational analysis of function learning, drawing on work on regression in machine learning and statistics. Using the equivalence of Bayesian linear regression and Gaussian processes, which provide a probabilistic basis for similarity-based function learning, we show that learning explicit rules and using similarity can be seen as two views of one solution to this problem. We use this insight to define a rational model of human function learning that combines the strengths of both approaches and accounts for a wide variety of experimental results.

A rational model of function learning

Every time we get into a rental car, we have to learn how hard to press the gas pedal for a given amount of acceleration. Solving this problem – which is an important part of driving safely – requires learning a relationship between two continuous variables. Over the past fifty years, several studies of *function learning* have shed light on how people come to understand continuous relationships (Carroll, 1963; Brehmer, 1971, 1974; Koh & Meyer, 1991; Busemeyer, Byun, DeLosh, & McDaniel, 1997; DeLosh, Busemeyer, & McDaniel, 1997; Kalish, Lewandowsky, & Kruschke, 2004; McDaniel & Busemeyer, 2005). It has become clear that people can learn and recall a wide variety of relationships, but demonstrate certain systematic biases that tell us about the mental representations and implicit assumptions that humans employ when solving function learning problems. For example, people tend to expect that relationships will be linear when extrapolating to novel examples (DeLosh et al., 1997), and find it more difficult to learn relationships that change direction than those that do not (Brehmer, 1974; Byun, 1995).

Several models have been developed to understand the cognitive mechanisms behind function learning. These models tend to fall into two different theoretical camps. The first includes *rule-based* theories (e.g., Carroll, 1963; Brehmer, 1974; Koh & Meyer, 1991), which suggest that people learn an explicit function from a given family, such as polynomials (Carroll, 1963; McDaniel & Busemeyer, 2005) or power-law functions (Koh & Meyer, 1991). This approach attributes rich representations to human learners, but has traditionally given limited treatment to how such representations could be acquired. A second approach includes *similarity-based* theories (e.g., DeLosh et al., 1997; Busemeyer et al., 1997), which focus on the idea that people learn by forming associations: if x is used to predict y, observations with similar x values should also have similar y values. This approach can be straightforwardly implemented in a connectionist architecture and thus gives an account of the underlying learning mechanisms, but faces challenges in explaining how people generalize so broadly beyond their experience. Most recently, hybrids of these two approaches have been proposed (e.g., Kalish et al., 2004; McDaniel & Busemeyer, 2005), with an associative learning process that acts on explicitly-represented functions.

Almost all past research on computational models of function learning has been oriented towards understanding the psychological processes that underlie human performance, or the steps by which people update and deploy their mental representations of continuous relationships. In this paper, we take a different approach, presenting a rational analysis of function learning in the spirit of Anderson (1990), Marr (1982), and Shepard (1987). Specifically, we start with an abstract representation of the problem to be solved and a handful of additional assumptions about the nature of continuous relationships, and then explore optimal solutions to the problem in light of these assumptions with the goal of shedding light on human behavior. This rational analysis provides a way to understand the relationship between the rule- and similarity-based approaches that have dominated previous work and suggest how they might be combined. Whereas hybrid models apply similarity-based learning to explicit rules, we offer a single foundation that supports both approaches, using a common set of commitments about learning and representation.

To understand the abstract problem that a function learner faces, we can turn to machine learning and statistics, where prediction in continuous domains – a problem familiarly known as regression – has been studied extensively. There are a variety of solutions to regression problems, but we focus on methods related to Bayesian linear regression (e.g., Bernardo & Smith, 1994), which allow us to make and test explicit claims about learners' expectations, using probability distributions. Bayesian linear regression is also directly related to a nonparametric approach known as Gaussian process prediction (e.g., Williams, 1998), in which predictions about the values of an output variable are based on the similarity between values of an input variable. We use this relationship to connect the two traditional approaches to modeling function learning, as it shows that learning rules that describe functions and specifying the similarity between stimuli for use in associative learning are not mutually exclusive alternatives, but rather two views of the same solution. We exploit this fact to define a rational model of human function learning that incorporates the strengths of both approaches.

The plan of this paper is as follows. First, we review several sets of empirical phenomena in function learning, both to provide background and to establish criteria by which different theories

of function learning can be judged. We then review past models of function learning, dividing them into rule-based, similarity-based, and hybrid approaches. Next, we introduce a new perspective on function learning in which rules and similarity can be expressed in a common framework, and describe a model that follows from this perspective. Finally, we evaluate different variations on our model against one another and previous models.

Phenomena in function learning

Past studies have taken diverse approaches to understanding how people learn relationships between continuous variables, but we will focus on four kinds of empirical phenomena that have been used in previous tests of function learning models (e.g., McDaniel & Busemeyer, 2005), or explicitly measure what kinds of relationships people implicitly believe to be more or less likely (Kalish, Griffiths, & Lewandowsky, 2007), or challenge many models of function learning (Kalish et al., 2004). Our decision to focus on the following phenomena is also motivated by their being relatively comparable, coming from similar experimental designs involving randomly-ordered, sequentially presented training stimuli, in the absence of informative cover stories or contextual information. In this section, we review these four kinds of phenomena, which we will later use to evaluate our own approach to explaining and understanding function learning.

Interpolation and learning difficulty

Some kinds of relationships are easier to learn than others. For example, increasing linear relationships tend to be easier to learn than decreasing linear relationships (Brehmer, 1971, 1976). Similarly, linear relationships are typically easier to learn than non-linear ones (Brehmer, 1974; Brehmer, Alm, & Warg, 1985; Byun, 1995; see Koh & Meyer, 1991 for a possible counterexample). Among non-linear relationships, people have more difficulty learning those that change direction (Brehmer, 1974; Brehmer et al., 1985; Byun, 1995). Cyclic relationships are especially difficult – but not impossible – to learn (Bott & Heit, 2004; Byun, 1995; Kalish, 2013). These systematic differences suggests that some relationships are subjectively simpler, more common, or

more straightforwardly represented than others, and the patterns given above dovetail with explicit human judgments about the probabilities of different kinds of relationships (Brehmer, 1974).

If the difficulty of learning a relationship reflects its mental representation, one can evaluate a model of function learning by comparing its average error rates to those of humans across several kinds of relationships. More precisely, if one orders several relationships by the average magnitude of errors that humans make when predicting y for x values that fall between past examples, i.e., interpolating, a good model should show the same ordering in its prediction error. For humans, these errors are influenced by many factors, such as the match or mismatch of cover stories to the available data, the number of training points, and presentation order (Byun, 1995), but we will focus on properties of the relationships themselves, which provide a simple basis for evaluating different theories of function learning. For instance, relationships in which y increases as a function of x tend to be easier to learn than functions in which y decreases as a function of x, which are in turn easier to learn than non-monotonic functions. For a summary of some qualitative properties of functions that contribute to differential learning difficulty for humans, see Busemeyer et al. (1997). In our own evaluation, we will use data from several studies that were gathered by McDaniel and Busemeyer (2005) and are summarized in Table 1.

Extrapolation

Studies that measure interpolation errors allow relationships to be ranked by how easy they are to learn, with implications for those relationships' subjective probability and consistency with humans' mental representations. Unfortunately, quite different models can show similar patterns of errors (given a limited set of relationship types) which constrains the amount one can learn from this approach. This and other limitations of interpolation-error studies have led some researchers to focus on how people extrapolate, or make judgments about points that are distant from those seen before. This approach gives a greater share of influence to learners' prior beliefs, and makes it possible to uncover patterns that are not reflected in interpolation error rates. To date, extrapolation-based studies of function learning are comparatively sparse, but have revealed several

biases in human learners. For example, people's extrapolation judgments follow linear patterns (DeLosh et al., 1997, but see Kalish et al., 2004), and more specifically tend toward functions with a positive slope and an intercept of zero (Kwantes & Neal, 2006). In one instance of this bias, when people are trained using data from a quadratic function, their average predictions fall between the true function and straight lines fitted to the closest training points.

Learning multiple relationships

The term "function learning" suggests that relationships between continuous variables – or at least the representations that people form of them – are functions, in that for a given value of the predictor x, there is a single valid prediction, or at least a range of predictions with a single most-likely value or mode. In reality, this is not always the case. For example, dose responses for drugs might have two or more patterns, depending on unobserved genetic factors or patient histories, and some hybrid cars have different relationships between pressure on the accelerator and the car's real acceleration, depending on whether or not the combustion engine is active. The world abounds with hidden mediators that can change the relationship between observable variables, and one might expect humans to be able to make judgments that reflect the presence of multiple underlying relationships. Consistent with this intuition, Lewandowsky, Kalish and Ngang (2002) found that fire fighters learn two distinct relationships between wind speed, ground slope, and the rate at which a fire spreads, depending on whether the fire is labeled as a standard forest fire, or a "back burn" fire set to mitigate damage from future fires. Lewandowsky et al. refer to this phenomenon as "knowledge partitioning", based on the idea that participants' knowledge of the relationship at hand is partitioned into distinct subsets based on context.

More recently, Kalish, Lewandowsky and Krushke (2004) conducted three experiments showing that people make judgments that demonstrate an implicit belief in the presence of multiple overlapping linear relationships, even when no contextual information was present, and in circumstances where the training data could be explained using a single non-linear relationship (see Figure 1 for examples).

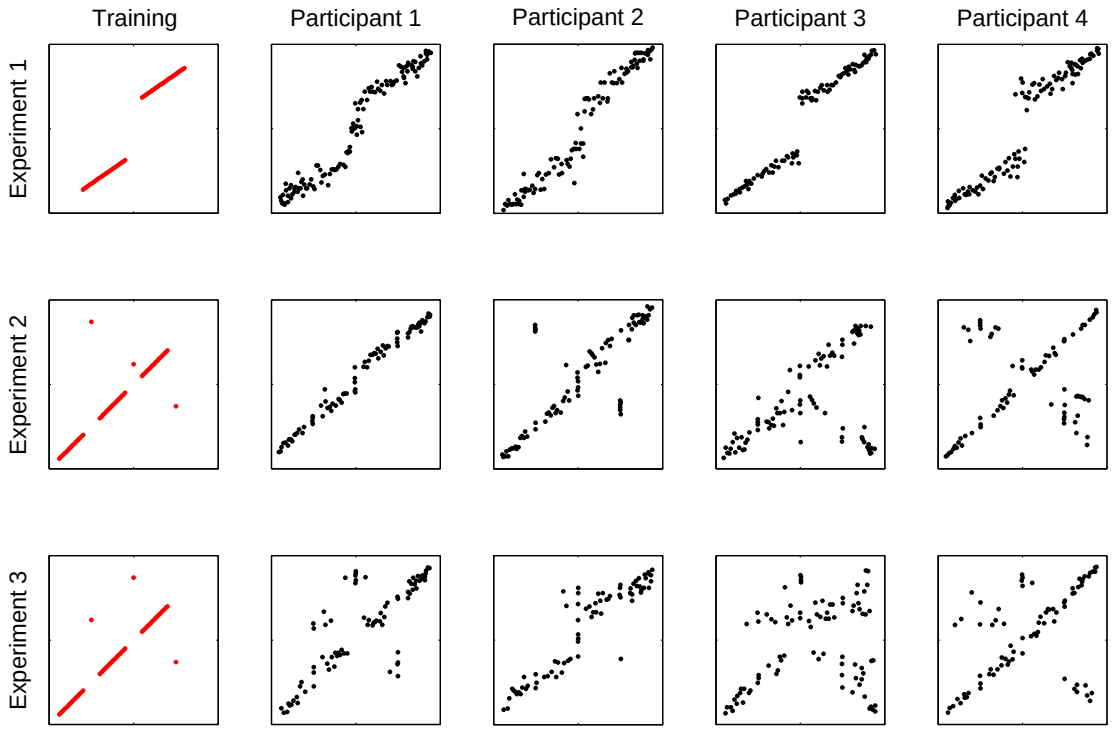

Figure 1. Training data and four participants' judgments for Experiments 1-3 in Kalish et al. (2004). Predictor variable values are plotted on the x-axes, with predicted variable values plotted on the y-axes.

Iterated learning

Iterated learning is an experimental method that was first developed for studying language evolution (Kirby, 2001), but it has more recently been applied to other phenomena, including function learning. In an iterated learning experiment, there are chains of learners where the first learner in each chain receives data, makes some inference on the basis of those data, and uses that inference to provide new data to the next learner in the chain. The data produced by each learner is the product of the data he or she receives and his or her inductive biases or expectations about the underlying relationship, item, or event. As the chain of learners grows longer, the influence of the learners' shared expectations eventually washes out the information carried by the data provided to the first learner. After enough iterations, the data carried forward in the chain reflect human expectations about what relationships are likely, rather than the data the first learner in the chain sees, providing useful information about how people represent and reason about the phenomena at

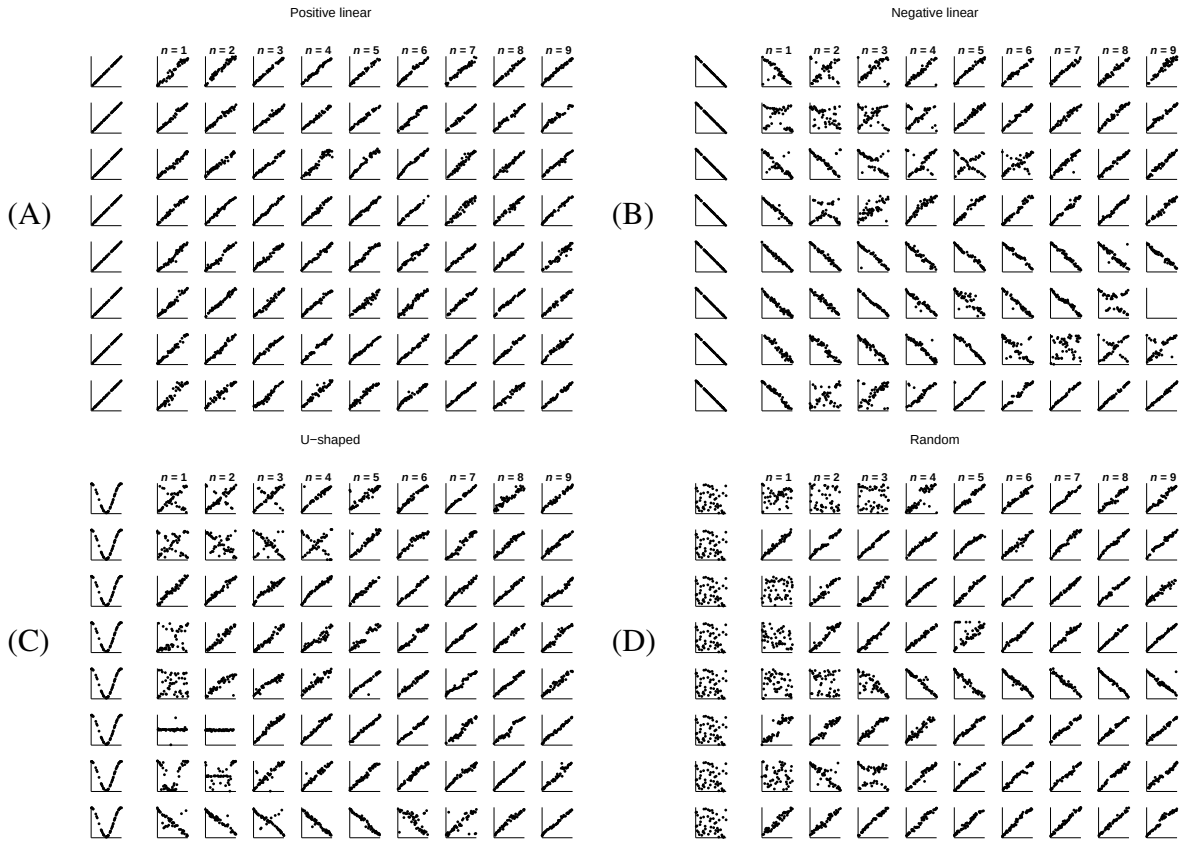

Figure 2. Plots of results from Kalish et al. (2007). (A) Positive linear initial data; (B) Negative linear initial data; (C) U-shaped initial data; (D) Random initial data.

hand (Kalish et al., 2007).

Figure 2 shows the results of a set of iterated function learning experiments conducted by Kalish et al. (2007). There were four conditions that differed in what data were given to the first participants in the chains. The *positive linear* (A) chains started with a linear relationship with a slope of one and an intercept of zero, the *negative linear* (B) chains started with a linear relationship with a slope of negative one and an intercept of zero, the *U-shaped* (C) chains started with data from a U-shaped relationship, and the *random* (D) chains started with a disorganized collection of points without any apparent underlying regularity. Kalish et al. (2007) found that the judgments of later participants tended to converge to a positive linear relationship with a slope of one and an intercept of zero regardless of the initial data. While these convergence results dovetail with past findings indicating that positive linear relationships are easier to learn, the intermediate

states of the chains provide a more detailed view of function learning. For example, learners tended to preserve negative linear relationships, consistent with the idea that people think these relationships are likely or plausible. Further, many learners were quick to infer the presence of multiple overlapping relationships, as when some participants interpreted noisy data as evidence for a negative linear relationship superimposed on a positive one.

Models of human function learning

The phenomena described in the previous section have inspired several theories and models of function learning, which can be organized into three classes: those based on rules or explicit functions, those based on associative or similarity-based learning, and hybrids that use explicit representations and associative learning. In this section, we review each class in turn, before discussing the extent to which each is consistent with the empirical results described above.

Representing functions with rules

Some of the earliest research into function learning postulates that people learn continuous relationships using explicitly-represented functions (Carroll, 1963). Carroll proposed that people assume a particular class of functions (such as polynomials of degree k) and use the available observations to estimate the parameters of those functions. The resulting representation allows people to generalize beyond the observed values of the variables involved. Consistent with the version of this hypothesis that Carroll advanced, people learned linear and quadratic functions better than random pairings of values for two variables, and extrapolated appropriately. Similar assumptions have guided subsequent work, which has explored the ease with which people learn different kinds of functions (e.g., Brehmer, 1974), and examined how well human responses are described by different forms of nonlinear regression (e.g., Koh & Meyer, 1991).

The advent of rule-based models precedes most of empirical results we consider, so it may be unsurprising that these models face some difficulty in explaining those results. Rule-based models do not show the flexibility in interpolation that human learners exhibit, and tend not to predict the order-of-difficulty found in interpolation studies (McDaniel & Busemeyer, 2005). Similarly, there

is evidence that rule-based models (such as Koh & Meyer, 1991) make extrapolation predictions that diverge from human judgments (DeLosh et al., 1997). Purely rule-based models make no provision for multiple overlapping relationships, and thus cannot account for knowledge partitioning effects (Kalish et al., 2004). By extension, their ability to explain Kalish, Griffiths, and Lewandowsky's (2007) iterated learning results is limited: while rule-based models might be able to explain long-run convergence to positive linear relationships, they do not anticipate participants' multimodal judgments.

Similarity and associative learning

Associative learning models propose that people do not learn relationships between continuous variables by explicitly learning rules, but instead forge associations between observed events and generalize based on the similarity of new variable values to old. The first model to implement this approach was the Associative Learning Model (ALM; DeLosh et al., 1997; Busemeyer et al., 1997), in which input and output arrays are used to represent a range of values for the variables between which the functional relationship holds. Presentation of an input activates input nodes close to that value, with activation falling off as a Gaussian function of distance, implementing a theory of similarity in the input space.

Learned weights determine the activation of the output nodes, which is a linear function of the activation of the input nodes. Weights are learned by gradient descent, where the local relationship between weights and errors is used to find new weights that reduce the squared error of the model's predictions. This process is repeated until the error can no longer be reduced. In practice, this approach performs well when interpolating between observed values, but poorly when extrapolating beyond those values, as it does not capture humans' ability to extrapolate in systematic, structured ways. As a consequence, Delosh et al. introduced the EXAM model, which constructs a linear approximation to the output of the ALM when selecting responses.

Similarity-based models have seen mixed success in explaining the range of empirical phenomena we describe above. In studies of interpolation and learning difficulty, similarity-based

models show similar patterns of interpolation errors to those of humans (McDaniel & Busemeyer, 2005). In the context of extrapolation, ALM does not address extrapolation but EXAM was developed with those results in mind and effectively captures the human bias toward linearity and predicts human extrapolations over a variety of relationships (McDaniel & Busemeyer, 2005), but without accounting for the human capacity for non-linear extrapolation (Bott & Heit, 2004). Like rule-based models, similarity-based models make unimodal predictions for any given x, and thus fail to account for knowledge partitioning results. This limitation also prevents EXAM from capturing some of the intermediate patterns that people produce in the iterated learning experiment.

Hybrid approaches

Several studies have explored methods for combining rule-like representations of functions with associative learning. One example of such an approach is the set of models explored in McDaniel and Busemeyer (2005). These models used the same kind of input representation as ALM and EXAM, with activation of a set of nodes similar to the input value. However, the models also feature a set of hidden units, where each hidden unit corresponds to a different parameterization of a rule from a given class, including polynomial, Fourier, and logistic functions. The values of the hidden units – corresponding to the values of the rules they instantiate – are combined linearly to obtain output predictions, with the weight of each hidden node being learned through gradient descent .

Another instance of a hybrid approach is the POLE model (Kalish et al., 2004), in which hidden units represent different linear functions and the weights from inputs to hidden nodes indicate which linear function should be used to make predictions for particular input values. Using this representation, the model can learn non-linear functions by identifying a series of local linear approximations, and can even model situations in which people seem to learn different functions in different parts of the input space. As a result, it is unique among the models we have discussed in its ability to match the bimodal response distributions discovered by Kalish et al. (2004).

Hybrid rule- and similarity-based models form a more heterogenous group than similarity-

and ruled-based models, with representatives including POLE (Kalish et al., 2004) and McDaniel and Busemeyer's (2005) connectionist implementations of rule-based models. POLE is set apart from the other models we have discussed by its ability to capture knowledge partitioning effects and it demonstrates a similar ordering of error rates to those of human learners (McDaniel, Dimperio, Griego, & Busemeyer, 2009). In its extrapolation predictions, however, there is evidence that it deviates from human performance (McDaniel et al., 2009). In an iterated learning design, POLE showed both convergence to positive linear relationships and some of the qualitative patterns that human learners demonstrate (depicted in Figure 3II), including transitional states with overlapping positive and negative linear relationships. McDaniel and Busemeyer's hybrid polynomial model – which performed better than the alternative hybrid models they considered – demonstrates an ordering of interpolation errors on different functions that aligns only roughly with human judgments (see Table 1), but its extrapolation predictions are consistent with human judgments from McDaniel and Busemeyer's studies (McDaniel & Busemeyer, 2005). Like rule-based models, this model offers unimodal predictions, and thus cannot account for knowledge partitioning phenomena, and has not been evaluated against iterated learning results.

Summary

We have reviewed a diverse set of models that accurately predict a variety of empirical phenomena in function learning. Despite their different commitments about how humans learn continuous relationships, a common theme of these models is an emphasis on the process by which function learning occurs. In the next section, we will take a fundamentally different view, focusing on the abstract problem of function learning and the forms that good solutions to that problem should take, rather than the process. This view complements past models rather than supplanting them, and we will demonstrate that it provides a common framework with which to understand and unify rule- and similarity-based approaches.

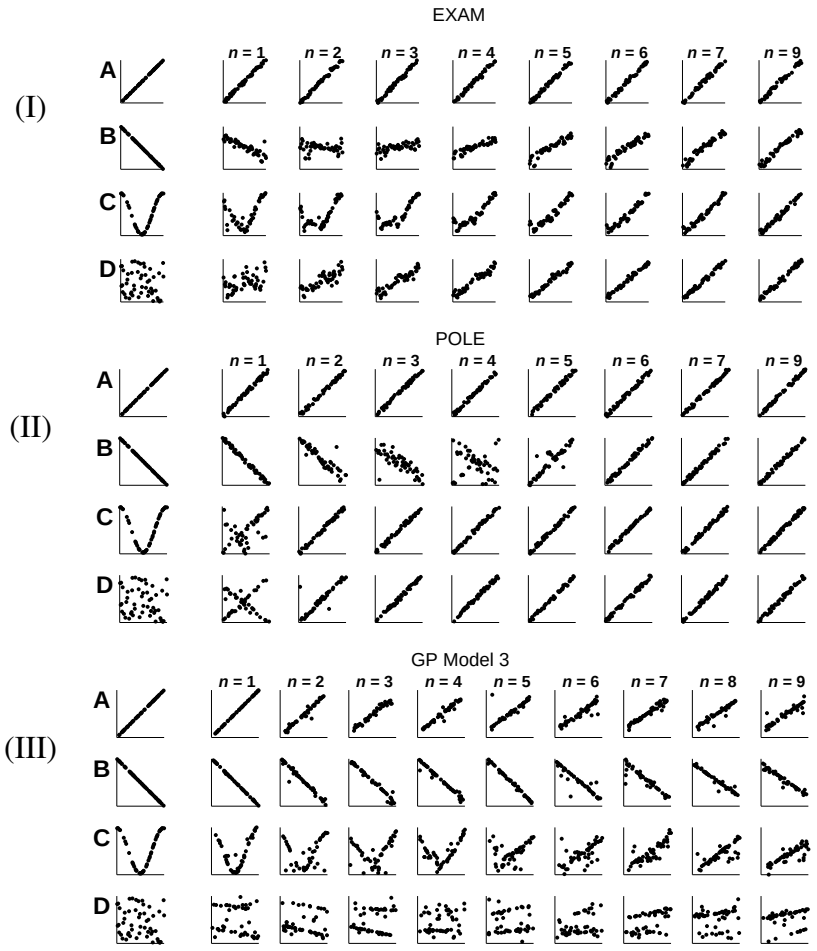

Figure 3. Model predictions for iterated learning data. A-D denote positive linear, negative linear, U-shaped, and random initial data, respectively. (I) Predictions from EXAM; (II) Predictions from POLE; (III) Mean function estimates from Model 3, removing noise.

Rational solutions to regression problems

The models outlined in the previous section all aim to describe the psychological processes involved in human function learning. In this section, we consider the abstract computational problem underlying this task, using optimal solutions to this problem to shed light on both previous models and human learning. Viewed abstractly, the computational problem behind function learning is to use a set of real-valued observations $\mathbf{x}_n = (x_1, ..., x_n)$ and $\mathbf{t}_n = (t_1, ..., t_n)$, to predict what y_{n+1} goes with a new x_{n+1}. Here, the y-values correspond to the underlying relationship, and the t-values are observations of y that have been obscured by additive noise, so $y_{n+1} = \mathbb{E}[t_{n+1}]$.

Following much of the literature on human function learning, we consider only one-dimensional relationships, but this approach generalizes naturally to the multi-dimensional case. In machine learning and statistics, this is referred to as a regression problem. In this section, we discuss how regression problems can be solved using Bayesian statistics, and how the result of this approach is related to Gaussian processes, a formalism with close ties to associative learning. Our presentation follows that in Williams (1998). See Appendix A for a more thorough treatment of the mathematical details.

Bayesian linear regression

Ideally, we would seek to solve our regression problem by using not just the observations of $\mathbf{x}$ and $\mathbf{t}$, but some prior beliefs about the probability of encountering different kinds of functions $f(\cdot)$ in the world. We can do this by applying Bayes' rule, with

$$p(f|\mathbf{x}_n, \mathbf{t}_n) = \frac{p(\mathbf{t}_n|f, \mathbf{x}_n)p(f)}{\int_{\mathcal{F}} p(\mathbf{t}_n|f, \mathbf{x}_n)p(f)df}. \tag{1}$$

Knowledge of which functions in the space of possibilities $\mathcal{F}$ are more likely to be the true function is captured by $p(f)$, the prior distribution. The probability of observing the values of $\mathbf{t}_n$ if f were the true function is given by the likelihood function $p(\mathbf{t}_n|f, \mathbf{x}_n)$, and the probability that f is the true function given the observations $\mathbf{x}_n$ and $\mathbf{t}_n$ is the posterior distribution $p(f|\mathbf{x}_n, \mathbf{t}_n)$. In most regression models, the likelihood is defined by assuming that any deviation from the true function is due to many independent sources of noise – more specifically, that t_i is Gaussian with mean $y_i = f(x_i)$ and variance σ_t^2. Predictions about the value of the function f for a new input x_{n+1} can be made by integrating over all functions in the posterior distribution,

$$p(y_{n+1}|x_{n+1}, \mathbf{t}_n, \mathbf{x}_n) = \int_f p(y_{n+1}|f, x_{n+1})p(f|\mathbf{x}_n, \mathbf{t}_n)df \tag{2}$$

where $p(y_{n+1}|f, x_{n+1})$ is a delta function placing all of its mass on $y_{n+1} = f(x_{n+1})$. Performing the integration outlined above can be challenging, but it becomes straightforward if we limit the

hypothesis space to certain specific classes of functions. If we take $\mathcal{F}$ to be all linear functions of the form $y = b_0 + xb_1$, then our problem takes the familiar form of linear regression. To perform Bayesian linear regression, we need to define a prior $p(f)$ over all linear functions. Since these functions are identified by the parameters b_0 and b_1, it is sufficient to define a prior over $\mathbf{b} = (b_0, b_1)$, which we can do by assuming that $\mathbf{b}$ follows a multivariate Gaussian distribution, which results in a posterior distribution over $\mathbf{b}$ that is also a multivariate Gaussian (see Bernardo & Smith, 1994). Linear transformations of Gaussian distributions are also Gaussian, so the predictive density (Equation 2) is also Gaussian, and the noise introduced between true values t and observations y simply adds to the variance of this distribution.

While considering only linear functions might seem overly restrictive, linear regression actually gives us the basic tools we need to solve this problem for more general classes of functions. Many classes of functions can be described as linear combinations of a small set of basis functions. For example, all kth degree polynomials are linear combinations of functions of the form 1 (the constant function), $x, x^2, ..., x^k$. Letting $\phi^{(1)}, \ ..., \ \phi^{(k)}$ denote a set of functions, we can define a prior on the class of functions that are linear combinations of this basis by expressing such functions in the form $f(x) = b_0 + \phi^{(1)}(x)b_1 + ... + \phi^{(k)}(x)b_k$ and defining a prior on the vector of weights $\mathbf{b}$. As long as the prior over weights is Gaussian, the same results apply as in the simple linear case.

Gaussian processes

Another approach to regression problems is to forgo any explicit representation of the underlying function and focus on making predictions. If our goal is merely to predict y_{n+1} using x_{n+1}, $\mathbf{t}_n$, and $\mathbf{x_n}$, we might simply define a joint distribution on $\mathbf{t}_{n+1}$ given $\mathbf{x}_{n+1}$ and find its expected value, which is equal to y_{n+1}, after conditioning on $\mathbf{t}_n$:

$$p(t_{n+1}|x_{n+1}, \mathbf{x_n}, \mathbf{t_n}) = \frac{p(\mathbf{t}_{n+1}|x_{n+1}, \mathbf{x_n})}{p(\mathbf{t}_n|x_{n+1}, \mathbf{x_n})}. \tag{3}$$

This equation expresses the problem of regression in very general terms, and may, at first glance, seem daunting to compute: it involves defining a joint distribution over all of the points observed so

far, as well as the joint distribution including the new, unknown point. Further, if we want to predict y_{n+1}, we must be able to take the expectation of this quotient. However, in some circumstances, the probability of t_{n+1} given x_{n+1}, $\mathbf{x_n}$, and $\mathbf{t}_n$ has a straightforward analytical solution, and an easily computed expectation. One such case, which will be our focus here, is when all $\mathbf{t}_{n+1}$ values are jointly Gaussian. In other words, $\mathbf{t}_{n+1}$ is distributed according to a single multivariate Gaussian, with dimensionality corresponding to the number of points under consideration. This is determined by its mean and covariance matrix, and once these are specified, we have a solution for Equation 3: the quotient has a closed form for multivariate Gaussians (see Rasmussen & Williams, 2006, for details). As we will see, assuming a jointly Gaussian distribution is not a strong constraint, and we can express a very broad set of relationships through our choice of means and covariances.

Both the mean vector and the covariance matrix are determined by the values of $\mathbf{x}$. Broadly speaking, the mean vector captures expectations about how the function looks in the absence of data, and the covariance matrix – or the *kernel function* that generates it – captures expectations about how points relate to one another. The covariance matrix entry for any pair of t-values (t_i, t_j) is given by a function $K(\mathbf{x}_i, \mathbf{x}_j)$, plus a diagonal matrix capturing the noisy relationship between the underlying values y_i and the observations t_i. we can Using this covariance matrix, we can obtain the distribution of t_{n+1} conditional on $\mathbf{t}_n$. The function $K(\cdot, \cdot)$, called the *kernel function*, can be chosen arbitrarily as long as the covariance matrix it produces is valid.

One common kind of kernel is a radial basis function, e.g.,

$$K(x_i, x_j) = \theta_1^2 \exp(-\frac{1}{\theta_2^2}(x_i - x_j)^2) \tag{4}$$

which leads to t values that are more strongly correlated when their corresponding x values are more similar, with the parameters θ_1 and θ_2 determining how quickly the correlation falls off as differences in x values increase. Other kernels are possible, including periodic functions such as

$$K(x_i, x_j) = \theta_3^2 \exp(\theta_4^2(\cos(\frac{2\pi}{\theta_5}[x_i - x_j]))) \tag{5}$$

indicating that values of y for which values of x are close relative to the period θ_3 are likely to be highly correlated.

This approach to prediction, in which a kernel function applied to $\mathbf{x}$ defines a normal distribution on t-values, is called a *Gaussian process*. A wide variety of kernel functions are possible, corresponding to varied commitments about which x values are likely to lead to similar t-values, making Gaussian processes a flexible way to solve regression problems.

Two views of regression

Bayesian linear regression and Gaussian processes appear to be quite different approaches. In Bayesian linear regression, a hypothesis space of functions is identified, a prior on that space is defined, and predictions are formed by averaging over the posterior distribution of y, while Gaussian processes simply use the similarity between different values of x, as expressed through a kernel, to predict correlations in values of y. It might thus come as a surprise that these approaches are equivalent.

Showing that Bayesian linear regression corresponds to Gaussian process prediction is straightforward. The assumption of linearity means that the vector $\mathbf{y}_{n+1}$ is equal to $\mathbf{X}_{n+1}\mathbf{b}$. Given normally distributed weights, it follows that $p(\mathbf{y}_{n+1}|\mathbf{x}_{n+1})$ is a multivariate Gaussian distribution with mean zero and covariance matrix $\mathbf{X}_{n+1}\mathbf{\Sigma}_b\mathbf{X}_{n+1}^T$. Bayesian linear regression thus corresponds to prediction using Gaussian processes, with this covariance matrix playing the role of $\mathbf{K}_{n+1}$ above (i.e., using the kernel function $K(x_i, x_j) = [1\ x_i][1\ x_j]^T$). Using a richer set of basis functions corresponds to taking $\mathbf{K}_{n+1} = \mathbf{\Phi}_{n+1}\mathbf{\Sigma}_b\mathbf{\Phi}_{n+1}^T$, i.e.,

$$K(x_i, x_j) = [1\ \phi^{(1)}(x_i)\ \ldots\ \phi^{(k)}(x_i)][1\ \phi^{(1)}(x_i)\ \ldots\ \phi^{(k)}(x_i)]^T, \tag{6}$$

where $\phi^{(1\ldots k)}$ are k arbitrary functions of x (Williams, 1998). It is also possible to show that Gaussian process prediction can always be interpreted as Bayesian linear regression, albeit with a potentially infinite number of basis functions. Just as we can express a covariance matrix in terms of its eigenvectors and eigenvalues, we can express a given kernel $K(x_i, x_j)$ in terms of its

eigenfunctions $\boldsymbol{\phi}$ and eigenvalues $\boldsymbol{\lambda}$, with

$$K(x_i, x_j) = \sum_{k=1}^{\infty} \lambda_k \phi^{(k)}(x_i)\phi^{(k)}(x_j) \tag{7}$$

for any x_i and x_j (Minh, Niyogi, & Yao, 2006). Thus, any kernel can be viewed as the result of performing Bayesian linear regression with a set of basis functions corresponding to its eigenfunctions, and a prior with covariance matrix $\boldsymbol{\Sigma}_b = \text{diag}(\boldsymbol{\lambda})$.

These results establish an important duality between Bayesian linear regression and Gaussian processes: for every prior on functions, there exists a corresponding kernel, and for every kernel, there exists a corresponding prior on functions. Bayesian linear regression and prediction with Gaussian processes are thus just two views of the same solution to regression problems.

Combining rules and similarity through Gaussian processes

The results outlined in the previous section suggest that, in the context of regression, learning using rules – as expressed in a Bayesian linear regression model – and generalizing based on similarity – as expressed in a Gaussian process's kernel function – are mutually compatible points of view. In this section, we briefly describe how previous accounts of function learning connect to these statistical models, and then use this insight to define a model of human function learning that combines the strengths of both approaches.

Reinterpreting previous accounts of human function learning

That idea of human function learning as a kind of statistical regression connects directly to Bayesian linear regression. Many rule-based models (e.g., Koh & Meyer, 1991; Carroll, 1963) can be framed in terms Bayesian linear regression while retaining all of their basic commitments and predictions. Similarly, the basic ideas behind Gaussian process regression (with a standard radial-basis kernel function) lie at the heart of similarity-based models such as ALM. In particular, ALM and the associative-learning component of EXAM implement cubic spline approximation (McDaniel & Busemeyer, 2005), which can be represented using Gaussian

processes (Rasmussen & Williams, 2006). Similarly, neural network approaches to similarity-based generalization are directly related to Gaussian processes, with some networks having a perfect mapping to a corresponding Gaussian process (Neal, 1994). Gaussian processes with radial-basis kernels can thus be viewed as implementing a simple kind of similarity-based generalization, predicting similar y values for stimuli with similar x values. The hybrid approach to rule learning taken by McDaniel and Busemeyer (2005) is also closely related to Bayesian linear regression. The rules represented by the hidden units serve as a basis set that specifies a class of functions, and applying penalized gradient descent on the weights assigned to those basis elements serves as an online algorithm for finding the function with highest posterior probability (MacKay, 1995).

Mixing functions in a Gaussian process model

The relationship between Gaussian processes and Bayesian linear regression suggests that we can define a single model that exploits both similarity and rules in forming predictions. We can do this by choosing a hypothesis space that covers a broad class of functions, including both those consistent with a radial basis kernel and those taking simple parametric forms. This is equivalent to modeling $\mathbf{y}$ as being produced by a Gaussian process with a kernel corresponding to one of a small number of types. Specifically, we assume that observations are generated by a function that is linear with positive slope, linear with negative slope, quadratic, or nonlinear but generally smooth. Figure 4 depicts samples from these individual kernels. This combination is one way to express the total prior over functions in Equations 1 and 2, with $p(f) = \sum_k p(f|k)P(k)$, where k represents a particular kernel in the set of four we have mentioned. For examples of functions that are likely under each of the different kernels, see Figure 4.

We do not claim that the specific kernels compose an exhaustive account of the relationships that people and learn and extrapolate from. Rather, we believe that people find these relationships especially easy to learn, and especially plausible or likely as explanations of data in the face of uncertainty, based on the results of Brehmer (1971), DeLosh et al. (1997) and Kalish et al. (2007).

A more complete account would include kernels that permit a wide variety of extrapolation

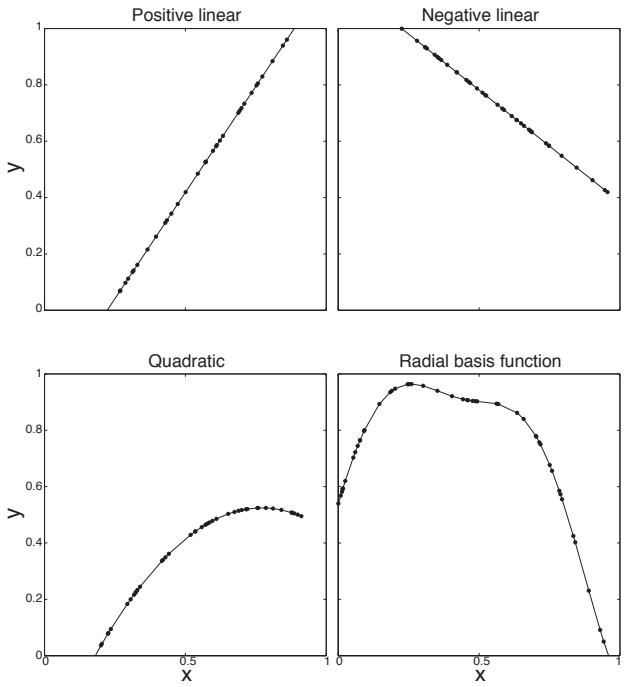

Figure 4. Samples from the four kernels that are combined in our models, reflecting the kind of relationship that each kernel favors.

patterns (e.g., Bott & Heit, 2004), but for the data we will consider such an expansion would add to the complexity of our models without substantially changing our predictions (see Lucas, Sterling, and Kemp (2012) for a demonstration of how Gaussian process models can be used to predict a variety of non-linear extrapolations). The probabilities of the different relationship types are defined by the vector $\boldsymbol{\pi}$. The relevant kernels are introduced in the previous sections (where "Nonlinear" corresponds to the radial basis kernel), with the positive and negative kernels having different means in their distributions over weights $\mathbf{b}$, taking mean intercepts and slopes of $[0\ 1]$, $[1\ -1]$ respectively. Using this Gaussian process model allows a learner to simultaneously make inferences about the overall type and specific form of the function from which their observations are drawn.

In developing this kind of model and selecting this particular set of priors – reflected in our choice of kernel functions – we are making explicit commitments about the inductive biases that shape human function learning. These include what types of relationships are more subjectively probable than others, and the more specific forms that relationships of a given type are likely to

take. Our model does not, however, commit to any specific process by which those biases shape people's inferences, which might resemble, for example, the associative mechanisms present in POLE or EXAM or an elaboration the hypothesis-testing framework offered by Brehmer.[1]

Basic tests of the Gaussian process model

In the remainder of the paper, we will evaluate our Gaussian process approach to function learning using each of the empirical phenomena we discussed earlier. First, following the approach taken in McDaniel and Busemeyer's (2005) review of computational models of function learning, we look at two quantitative tests of Gaussian processes as an account of human function learning: reproducing the order of difficulty of learning functions of different types, and extrapolation performance. As indicated earlier, there is a large literature consisting of both models and data concerning human function learning, and these simulations are intended to demonstrate the potential of the Gaussian process model rather than to provide an exhaustive test of its performance. See Appendix B for a summary of the parameters in our model, and Appendix C for a description of the procedures used to generate model predictions.

Difficulty of learning

As discussed above, one important measure of a theory of human function learning is its ability to account for the relative difficulty people have in learning different kinds of relationships. Table 1 is an augmented version of results presented in (McDaniel & Busemeyer, 2005) which compared several models' prediction errors to humans' errors when learning a range of functions. Each entry in the table is the mean absolute deviation (MAD) of human or model responses from the actual value of the function, evaluated over the stimuli presented in training. The MAD provides a measure of how difficult it is for people or a given model to learn a function. The data reported for each set of studies are ordered by increasing MAD (corresponding to increasing

[1] In obtaining predictions from our model, we use sampling methods that are described in Appendix C. There has been recent work supporting the idea that sampling may explain the inferences that humans make in many domains (Griffiths, Vul, & Sanborn, 2012), but our predictions are not coupled to any inference procedure and we do not have data distinguishing between different mechanistic or implementation-level accounts.

difficulty). In addition to reproducing the MAD for the models in (McDaniel & Busemeyer, 2005), the table has been expanded to contain the MADs exhibited by seven Gaussian process (GP) models trained on the target functions.

The seven GP models incorporated different collections of kernel functions by adjusting their prior probabilities. The most comprehensive model includes the $\{\text{Positive Linear}, \text{Negative Linear}, \text{Quadratic}, \text{Nonlinear}\}$ set of kernel functions, assigning them prior probabilities proportional to 8, 1, 0.1, and 0.01, respectively.[2] Six other GP models were examined by assigning certain kernel functions zero prior probability and re-normalizing the remainder so that the prior probabilities summed to one. The seven distinct GP models are presented in Table 1 are labeled by the kernel functions to which they assign non-zero probability, under the header "Model 1". Models 2 and 3, which are extensions that account for knowledge partitioning phenomena, are discussed below. The kernels include *Linear* (including both positive and negative linear functions), *Quadratic* (second-order polynomial functions), *RBF* (nonlinear relationships, fit by a radial basis function kernel), *LQ* (linear and quadratic), *LR* (linear and RBF), *QR* (quadratic and RBF), and *LRQ* (linear, quadratic, and RBF). The MAD for each function from McDaniel and Busemeyer (2005) is reported for each model in Table 1, along with human MADs. The last three rows of Table 1 give the correlations between human and model performance across functions, expressing quantitatively how well each model captured the pattern of human function learning behavior. All of the GP models perform well, with every model (except for the Linear and *LQ* models) providing a closer match to the human data than any of the models considered by McDaniel and Busemeyer (2005).

Extrapolation performance

Predicting and explaining people's capacity for extrapolation to novel stimuli is another key criterion for judging models of function learning. In Table 2, we compare mean human predictions for linear, exponential, and quadratic functions (from DeLosh et al., 1997) to those of several

[2]The selection of these values was guided by results indicating the order of difficulty of learning functions of these different types for human learners, but we did not optimize π with respect to the criteria reported here.

Table 1

Difficulty of Learning Results Based on Experiments Reviewed in McDaniel and Busemeyer (2005).

			Hybrid models			Model 1							Gaussian process expert models	
Function	Human	ALM	Poly	Fourier	Logistic	Linear	Quad	RBF	LR	LQ	RQ	LRQ	GPE (Model 2)	Local GPE (Model 3)
Byun (1995, Expt 1B)														
Linear	.20	.04	.04	.05	.16	.00038	.02	.032	.0044	.00063	.03	.0021	.011	.015
Square root	.35	.05	.06	.06	.19	.044	.039	.043	.047	.043	.05	.045	.038	.044
Byun (1995, Expt 1A)														
Linear	.15	.10	.33	.33	.17	.0011	.021	.033	.0037	.00092	.03	.0028	.0079	.013
Power, pos. acc.	.20	.12	.37	.37	.24	.066	.021	.053	.067	.055	.033	.065	.053	.058
Power, neg. acc.	.23	.12	.36	.36	.19	.044	.039	.037	.046	.042	.048	.045	.049	.045
Logarithmic	.30	.14	.41	.41	.19	.07	.051	.046	.073	.067	.063	.073	.07	.066
Logistic	.39	.18	.51	.52	.33	.11	.11	.079	.11	.11	.11	.11	.11	.11
Byun (1995, Expt 2)														
Linear	.18	.01	.18	.19	.12	.00015	.017	.020	.0023	.00038	.022	.0017	.011	.0081
Quadratic	.28	.03	.31	.31	.24	.26	.041	.044	.14	.0084	.082	.055	.083	.072
Cyclic	.68	.32	.41	.40	.68	.30	.29	.18	.29	.29	.28	.17	.31	.31
Delosh, Busemeyer, & McDaniel (1997)														
Linear	.10	.04	.11	.11	.04	.00021	.019	.026	.0042	.00049	.026	.0025	.0052	.0089
Exponential	.15	.05	.17	.17	.02	.05	.039	.033	.05	.047	.05	.05	.048	.049
Quadratic	.24	.07	.27	.27	.11	.26	.053	.056	.21	.010	0.10	.068	.11	.073
Overall correlation between human and model performance														
Linear	1.0	.83	.45	.45	.93	.66	.93	.93	.78	.91	.92	.89	.92	.93
Rank-order	1.0	.55	.51	.51	.77	.72	.82	.80	.71	.69	.80	.74	.81	.79
Mean within-experiment correlation between human and model performance														
Linear	1.0	.99	.97	.97	.90	.91	.98	.96	.96	.74	.61	.96	.99	.97

Note: Rows correspond to functions learned in experiments reviewed in McDaniel and Busemeyer (2005). Columns give the mean absolute deviation (MAD) from the true functions for human learners and different models (Gaussian process models with multiple kernels are denoted by the initials of their kernels, e.g., LQR = Linear, Quadratic, and Radial Basis Function). Human MAD values represent sample means (for a single subject over trials, then over subjects), and reflect both estimation and production errors, being higher than model MAD values which are computed using deterministic model predictions and thus reflect only estimation error. The last three rows give the correlations of the human and model MAD values, including linear and rank-order correlations across all experiments and mean linear correlations on an experiment-by-experiment basis, providing an indication of how well the model matches the difficulty people have in learning different functions.

models described in McDaniel and Busemeyer (2005), as well as each of the Gaussian process models we describe above and two model extensions that we will describe below. While none of the GP models produce quite as high a correlation as EXAM on all three functions, all but the Linear and LR models make predictions that correspond closely with human judgments. It is notable that this performance is achieved with the same parameters that were used for the difficulty of learning data (see Appendix B for details), while the predictions of EXAM were the result of optimizing two parameters for each of the three functions.

Figure 5 displays mean human judgments for each of the three functions, along with the predictions of an extended Gaussian process model we discuss below, which incorporates Linear, Quadratic, and Nonlinear kernel functions. The regions to the left and right of the solid black lines represent extrapolation regions, containing input values for which neither people nor the model were trained. Both people and the model extrapolate nearly optimally on the linear function, and reasonably accurately for the exponential and quadratic function. However, there is a bias towards a linear slope in the extrapolation of the exponential and quadratic functions, with extreme values of the quadratic and exponential function being overestimated. Quantitative measures of extrapolation performance are shown in Table 2, which gives the correlation between human and model predictions for EXAM (DeLosh et al., 1997; Busemeyer et al., 1997) and the seven GP models.

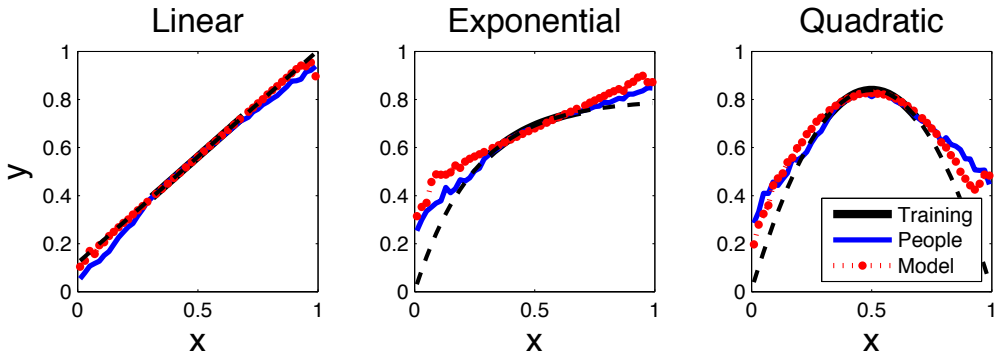

Figure 5. Extrapolation performance, with mean predictions on linear, exponential, and quadratic functions for human participants from Delosh, Busemeyer and McDaniel (1997) and a mixture of Gaussian process experts (Model 3; see text). Training data were presented in the region spanned by a solid black line, , and extrapolation performance was evaluated outside this region, with the true function represented by dashed lines.

Table 2
Linear Correlations Between Human and Model Predictions for Extrapolation Regions.

Model	Linear	Exponential	Quadratic
EXAM	.999	.997	.961
Model 1, Linear	.996	.972	.277
Model 1, Quadratic	.986	.973	.921
Model 1, RBF	.996	.989	.921
Model 1, LQ	.996	.972	.942
Model 1, LR	.996	.972	.363
Model 1, RQ	.985	.967	.941
Model 1, LRQ	.996	.971	.513
Model 2	.996	.981	.955
Model 3	.996	.982	.957

Note: Gaussian process models with multiple kernels are denoted as in Table 1.

Summary

We have shown that our model accounts well for the relative difficulty with which people learn different kinds of relationships, and how they extrapolate from limited training data. More complex phenomena, such as knowledge partitioning and the multiple overlapping relationships it entails, require more complex models. The next section addresses these phenomena, and describes a straightforward extension of our Gaussian process model to accommodate the possibility of multiple relationships while still explaining human interpolation and extrapolation behavior.

Extending the Gaussian process model beyond single relationships

In most models of function learning, including the Gaussian process-based models described above, it is assumed that people learn a single relationship between a variable and its predictors. There might be a complex, non-linear relationship between $\mathbf{x}$ and $f(\mathbf{x})$, but for a single value of $\mathbf{x}$, $f(\mathbf{x})$ is always unimodal and relationships are never compositions of other relationships. We have mentioned that this assumption fails to describe many real relationships, and, as knowledge partitioning results show, it also fails to explain human behavior.

Of the models we have described, only the POLE model (Kalish et al., 2004) makes

predictions that are consistent with knowledge partitioning phenomena, doing so by appealing to the mental representations and processes people use when learning functions. We will show that a rational analysis of function learning leads to a similar set of predictions. In many real-world situations, two variables x and y will relate to one another in different ways, depending on context. If y depends on w in addition to x, i.e., the true function is $y = f(x, w)$, and w is not observable, the apparent relationship between x and y may have discontinuities, and it may not be a function at all, having multiple values of y for a given x . We previously discussed examples of such relationships, including acceleration in hybrid cars and dose-response curves in a patient population. Other examples of hidden mediators include the relationship between brake pressure and acceleration, mediated by surface slipperiness, and the relationship between the temperature of a material and its malleability, mediated by its unobserved crystal structure, as with the temper of a piece of metal. With these intuitions in hand, we will now describe how our model may be extended to reflect them.

Mixtures of Gaussian process experts

We extended our Gaussian process model (Model 1), to capture the assumption that each point belongs to one of an unknown number of underlying relationships. Clearly, there is no fixed bound on the number of relationships that might obtain between x and y, but one would expect that fewer relationships should be more plausible than more, as a matter of simplicity or parsimony (Chater & Vitanyi, 2003). There are multiple ways to express this intuition formally, but one obvious choice is to allow points to be divided into arbitrary partitions, assigning each partition a probability using a Chinese Restaurant Process prior (Aldous, 1985), which has previously been used in rational analyses of categorization (Anderson, 1991; Sanborn, Griffiths, & Navarro, 2010).

Under this prior, the likelihood that a new (x, y) pair will be assigned to an existing relationship is proportional to the number of other points that participate in that relationship, and the likelihood that it will be assigned to a new relationship is proportional to a parameter α. More

precisely, the probability that the i^{th} point's relationship r_i will be k is

$$\Pr(r_i = k) = \begin{cases} \dfrac{n_k}{i+\alpha} & \text{if } n_k > 0, \\ \dfrac{\alpha}{i+\alpha} & \text{if } n_k = 0 \end{cases} \tag{8}$$

where n_k is the number of points already participating in relationship k. The likelihood of the data under a given partition is determined by how likely the ensemble of y values is, given the nature of the relationships they participate in and their corresponding x values. This conceptually straightforward extension from Gaussian processes to a mixture of Gaussian processes will be called Model 2. We might also wish the capture the intuition that (x, y) pairs that have similar x values are more likely to participate in the same relationships – in other words, relationships tend to be locally smooth and unimodal. This expectation can be built into the model by assuming that the likelihood that a point belongs to a partition is determined in part by its closeness to current members, represented using the $\mathbf{x}$-value's likelihood under a Gaussian distribution based on existing members. This last model, Model 3, is an example of a mixture of experts (Jacobs, Jordan, Nowlan, & Hinton, 1991; Erickson & Kruschke, 1998; Kalish et al., 2004), an approach that has been applied to Gaussian processes in the past (Rasmussen & Ghahramani, 2002; Meeds & Osindero, 2006). As with Model 1, Models 2 and 3 can be interpreted in terms of Bayesian linear regression or Gaussian processes, where every Gaussian process kernel for every expert can be represented as a linear regression model, albeit, as before, with a potentially infinite number of features. See Figure 6 for samples of the kinds of relationships that the mixture of Gaussian process experts (henceforth Model 3) favors.

Knowledge partitioning

Before applying Models 2 and 3 to knowledge partitioning phenomena, we evaluated them against the same difficulty-of-learning and extrapolation results with which we assessed our original Gaussian process models. As with the earlier models, we used the same parameters for all of the experiments, and obtained close fits to human judgments, summarized in Tables 1 and 2 (see

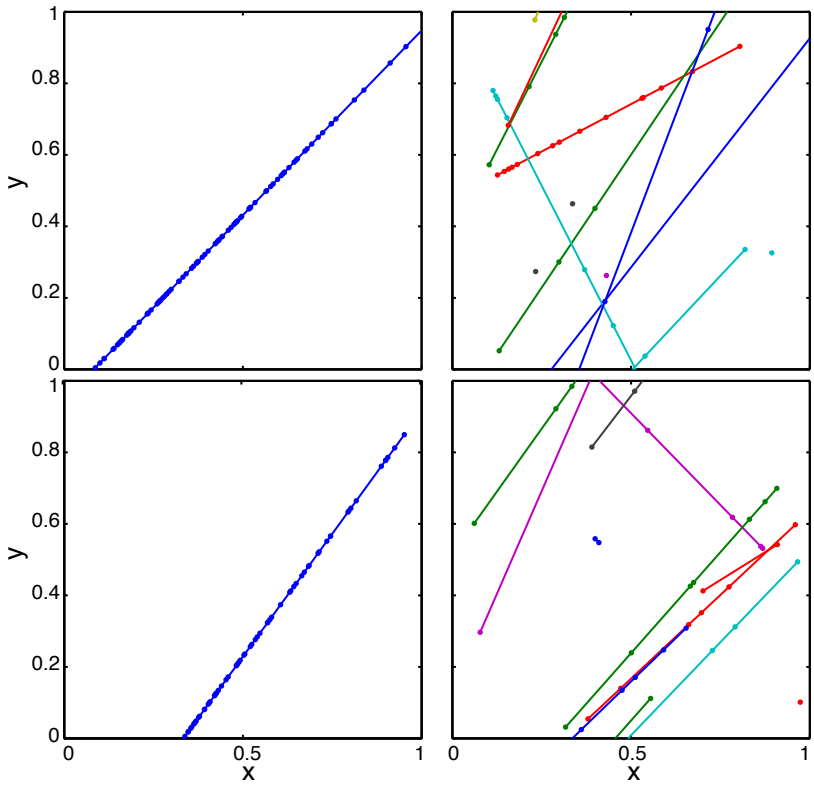

Figure 6. Samples from Model 3. The left plots show samples drawn from an infinite mixture of experts with $\alpha = .1$, favoring a small number of distinct relationships. The right plots show samples drawn from a mixture with $\alpha = 10$, favoring a large number of distinct relationships. Because of the diffuse prior over locations in the x-axis in Model 3, randomly-drawn samples tend not to concentrate in x and thus look similar to samples drawn from Model 2.

Appendix B for details of parameters and fits). We also plotted predictions for Model 3 against mean human judgments in the extrapolation experiments in Figure 5. In general, Models 2 and 3 performed as well as any other model, and better than the majority of the alternatives.

To gauge the extent to which the models' predictions are consistent with knowledge partitioning phenomena, we obtained individual predictions from twelve participants in Kalish et al.'s (2004) studies, four per experiment.[3] Each experiment included training points and interpolation regions that were designed to elicit multiple modes in y for a given x. For example, in Experiment 1, there was a gap between two partial linear functions with the same slope and different intercepts. Many participants made judgments in the gap that matched both functions,

[3]The full data set was not available, but the combined distribution of judgments for the 12 participants was consistent with the overall distribution reported in Kalish et al. (2004).

leaving a bimodal response distribution. Like Kalish et al., we focus on showing that our model captures the bimodal responses of the participants, and gives a posterior distribution that matches the distribution of actual judgments.

The results are summarized in Figure 7, comparing Models' 1, 2, and 3 predicted probabilities of different y values to those given by participants. Model 1 predicts the aggregate trend in Kalish et al.'s Experiment 1, but cannot explain the discontinuities exhibited by two of the participants shown in Figure 1) or the multiple modes evident in participants' judgments for Experiments 2 and 3. In contrast, Models 2 and 3 predict the multiple relationships will be inferred. Model 3, being sensitive to the proximity of points, is more likely than Model 2 to group points into local relationships, as is apparent in its predictions for Experiment 1. We used a single prior distribution across the different experiments and participants, but the individual differences in Figure 1 are readily explained in terms of different participants having different inductive biases. Future work, with more extensive within-subjects data, would permit us to test our model as a framework for understanding how inductive biases vary between individuals.

Iterated learning

As a final measure of Gaussian process models of function learning, we compared their predictions to human judgments in the iterated learning experiments of Kalish et al. (2007). As mentioned earlier, iterated learning designs involve a chain of learners in which each individual observes data, makes inferences from those data, and uses those inferences to provide data to the next learner in the chain. For function learning specifically, each observation is an (x, y) pair, and the data that a learner passes forward is a subset of his or her y-predictions for new x-values. Ideally, these judgments would reflect samples from the inferred underlying function, with variance attributable only to uncertainty about that function, and, potentially, inferred noise around that function. In practice, however, participants' judgments are subject to errors in perception and in recording their judgments, as well as varying degrees of motivation and attention. Rather than attempting to model these factors – which are underdetermined – we chose to apply our

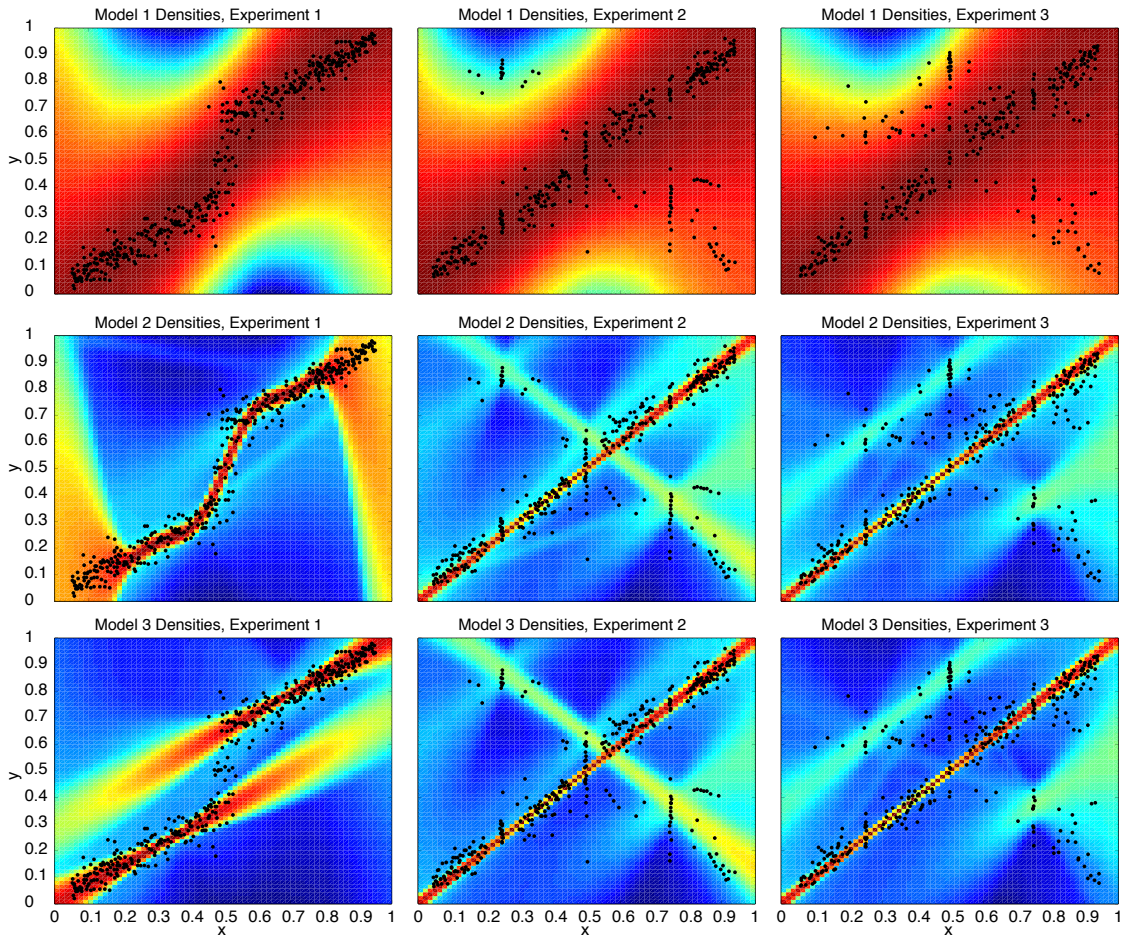

Figure 7. Plots comparing human judgments in Experiments 1-3 of Kalish et al., (2004) to the predictions of Models 1, 2, and 3. The points represent individual human judgments, aggregated over four individuals for whom data were available, while the colors represent log probability densities, with hotter colors representing higher probabilities.

mixture-of-experts model to the same tasks that human faced as-is, looking for the same qualitative patterns that human learners demonstrated. As in Kalish et al.'s experiments, we ran chains in which the first iteration's observations, or the *initial data*, were drawn from four functions, including positive linear, negative linear, U-shaped, and random functions. For each subsequent round, the model used 50 predictions generated from the previous round, like the human learners.

The human learners' judgments revealed several broad patterns, shown in Figure 2, which we used as the basis for our evaluation, including: (1) given positive linear initial data, judgments were consistently positive linear over successive rounds; (2) a shift toward positive linear functions for

the negative linear, U-shaped, and random initial data, with transitional states reflecting uncertainty or inferences to high noise or multiple overlapping relationships – in almost all chains, there are intermediate states that deviate from any simple, well-formed function; and (3) greater stability and slower transitions in the negative linear case than in the U-shaped and random cases.

Figure 3III shows that Model 3 demonstrates each of these features. Like many human learners and the POLE and EXAM models (Figure 3I and II), it preserved positive linear relationships, with small deviations from a 0-intercept 1-slope relationship that are due to our treatment of out-of-range samples: when the model samples y values that are greater than 1 or less than 0, those values are resampled, leading to a slight flattening of the slope. A policy of converting out-of-range samples by replacing them with the most extreme value would reduce this effect. Like human chains and the POLE model's predictions, but not EXAM's, iterations following U-shaped initial data included cases of overlapping positive and negative relationships. Like several human learners, random initial data led to the GP model to offer overlapping, weakly sloped linear relationships before shifting towards a single positive linear relationship. Finally, like human learners, the GP model tended to preserve the negative linear relationships more than U-shaped and disordered relationships. The most salient difference between the GP Model 3 and human learners is its slower convergence to positive linear relationships.

There are several ways in which we might account for this difference in convergence rates. First, our priors over types of relationships were not fitted to human behavior, and one more strongly favoring positive linear relationships – or a lower variance in the distribution of slopes – would naturally lead to faster convergence. Second, a more nuanced view of noise would be consistent with the differences in convergence rates. For example, our model assigns a very low probability to "random" relationships, in which points have very high variance, whereas participants might expect that some points are anomalies, analogous to equipment failures. Third, the rapid convergence of human chains might be explained in part by differences between individual human learners. For example, specific individuals might have stronger expectations that relationships are positive and linear, and believe more strongly that their observations are only noisy

reflections of the underlying relationship. As with individual differences in knowledge partitioning, all of these possibilities could be explored using within-subjects data.

General Discussion

Function learning is one of the core inductive problems that we encounter every day, arising whenever we need to learn the relationship between two continuous variables. Models of function learning have explained the human ability to solve this problem in terms of different cognitive mechanisms, such as inducing rules or generalizing on the basis of similarity. We have shown that these different cognitive mechanisms correspond to different strategies for solving the abstract computational problem of regression, and that both can be expressed as special cases of a Bayesian solution to this problem based on Gaussian processes. This perspective helps to reveal the commonalities between these different mechanisms, and to define models that combine their strengths. The resulting models provide a good fit to human data, performing similarly to the best mechanistic accounts, and provide a way to transparently identify the inductive biases that guide human learners in function learning tasks.

In our introduction, we stated that our model is intended to complement, rather than replace, existing accounts of function learning: we focus on the inductive biases that shape function learning, rather that the processes by which it occurs. In the remainder of this paper, we will discuss the relationship between these levels of analysis, the project of identifying human inductive biases, and some of the ways in which our work could be extended.

The roles of models at different levels of analysis

Our focus in this paper has been on understanding human function learning by identifying the underlying computational problem and the assumptions that seem to yield parallels between optimal solutions to this problem and human behavior. This approach is in the spirit of the approach of rational analysis laid out by Anderson (1990), yielding an explanation of behavior that lies at what Marr (1982) termed the “computational level”. The results of this investigation are quite different from those yielded by a more traditional modeling approach operating at what Marr (1982)

termed the "algorithmic level" and focusing on identifying the cognitive mechanisms underlying human behavior. The previous models of function learning we have discussed in this paper are defined at this level, making claims about the aspects of human memory and reasoning that contribute to their performance on function learning tasks.

The focus on the computational level establishes a clear set of goals for our model. First, we are not trying to define the single best model of human performance on function learning tasks, because our computational-level model is not in competition with algorithmic-level models. It is entirely possible for our computational-level analysis to be correct, and for it to be executed at the algorithmic level by cognitive mechanisms that resemble existing psychological process models. In this case, we would expect both kinds of models to fit well (and possibly the process models to fit better, since they will capture idiosyncrasies of behavior due to the way in which the computational-level solution is carried out). Our goal is to show that the computational-level solution we have proposed does a good job of capturing human behavior, and existing algorithmic-level models provide a good yardstick against which to measure this performance.

Second, a key part of our contribution is theoretical. We have shown that algorithmic-level mechanisms that seem quite different can in fact be captured in a single theoretical framework at the computational level, and that this leads to new ways of thinking about combining the strengths of these approaches. This kind of contribution has a precedent in other work examining aspects of cognition at different levels of analysis: Ashby and Alfonso-Reese (1995) showed that exemplar and prototype models of categorization could both be viewed as strategies for solving the problem of density estimation that arises when categorization is viewed from the perspective of Bayesian inference. This demonstration of a common underlying computational-level problem (and connections to ideas in statistics) provides the foundation for recent work on rational models of categorization that can interpolate between exemplar and prototype representations (Sanborn et al., 2010). We view our analysis as making a similar contribution for the case of function learning, providing an explicit link between existing cognitive models and ideas from statistics that leads to new ways of understanding human behavior. A probabilistic approach also provides a basis for

understanding a broader range of phenomena, including not just patterns of interpolation and extrapolation judgments. For example, one can use explain the influence by linguistic and contextual information on function learning (Byun, 1995) in terms of priors, and understand people search for new information (Borji & Itti, 2013) or benefit from different kinds of instruction (Lindsey, Mozer, Huggins, & Pashler, 2013).

Capturing human inductive biases

In inductive problems, such as function learning, the right answer is underdetermined by the available data. This means that doing a good job of solving the problem requires having good inductive biases – those factors other than the data that lead a learner to favor one hypothesis over another (Mitchell, 1997). When viewed from the abstract computational level, the key challenge in explaining human inductive inference is characterizing our inductive biases. Bayesian models of cognition make this task particularly clear, as the inductive biases of these models are expressed through the choice of hypothesis space and the prior on hypotheses.

In function learning, the characterization of the inductive biases of a learner is particularly clear: it corresponds to a prior distribution on functions. As we have discussed, defining a prior distribution on functions is challenging, since there are uncountably many possible functions, dependent on an unbounded number of latent variables. The Gaussian process models we have explored provide a succinct way of expressing priors on functions that is nonetheless extremely flexible in the range of distributions that it allows, and thus provide a powerful tool for exploring human inductive biases for function learning. We can express the assumptions behind this prior in terms of a kernel function, which captures the similarity between stimuli, in terms of a set of basis functions, which express a representation of these stimuli, or through samples from the resulting distribution over functions, providing three different ways to indicate the inductive biases that a learner has.

Being able to characterize human inductive biases in terms of a probability distribution over functions also makes it straightforward to make automated learning systems that are guided by the

same inductive biases. We can easily take the prior assumed by our Gaussian process models and use it as a component of Gaussian process models used in machine learning or statistics. This provides a natural bridge between human and machine learning, and an opportunity to explore whether using human inductive biases improves the operation of automated systems as well as to develop automated systems that make inferences that are more comprehensible to human users.

Limitations and future directions

The models we have explored cover a wide range of results from the literature on human function learning, but there are still phenomena that they cannot capture and aspects of human performance that lie outside the considerations that normally inform a computational-level analysis. Addressing these limitations creates some interesting directions for future work.

A basic omission in the formulation of our model is that it is unable to learn cyclic functions. Since these functions are learnable by people (although with significant difficulty Bott & Heit, 2004; Byun, 1995), this is a weakness that should be addressed. It is straightforward to incorporate a capacity to learn cyclic functions by including a periodic kernel in the mixture of kernels. Incorporation of this additional kernel – with an appropriately low mixture weight – would not change the predictions of the model for non-cyclic functions appreciably. We judged the corresponding increase in the complexity of the model to outweigh the value of capturing these additional phenomena.

The fact that people can learn cyclic functions raises another interesting question: Can we build an exhaustive summary of the kinds of relationships that people can learn? In the context of our models, this becomes a question of what kinds of functions have support in people's prior distributions, or what set of kernels should be included in the mixture. Existing results support the inclusion of a relatively small set of kernels – essentially, those that we consider plus a periodic kernel for cyclic functions.

Another issue, also related to our prior over kernels, is that we chose a distribution strongly favoring linear relationships. Is this prior consistent with the idea that a rational analysis should use

diffuse priors that capture the statistical structure of the environment (Anderson, 1990)? It is a shortcoming of the current work that we cannot be certain, but we believe that a linearity-biased prior is better than alternatives. In function learning, it is not realistic to directly measure the statistical structure of the environment, i.e., what functions are truly more or less common: doing so would depend on knowing what combinations of variables are salient to human observers over long periods of time, including, perhaps, our evolutionary history. Further, any census of functions would reflect the cognitive and attentional biases of the people who would conduct it. In the absence of ground truth about the frequencies of functions, we believe that the best approach is to look at what relationships people think are more common, using both direct and indirect measures. Previous studies, including many that we have not evaluated here (see Busemeyer et al.,1997 for a summary, and Little and Shiffrin, for evidence that people infer linear relationships given very noisy data) support the idea that linear relationships are thought to be more common. Among these are results showing that people say that linear relationships occur much more frequently than non-linear ones, and showing that people tend to offer linear relationships when prompted in the absence of data or informative context (Brehmer, 1974). Even if we set aside these results, it seems a case can be made that linear functions are indeed very common in situations the matter to humans. Under usual (e.g., non-relativistic) conditions, relationships between mass, force, acceleration, velocity, distance, and time can be expressed as collections of linear relationships, and many physical objects have broadly similar shapes at different scales, implying that an object's height is a roughly linear function of its width, for example.

Our focus on the abstract computational-level problem underlying function learning and the nature of ideal solutions to that problem means that there are aspects of human performance that our models cannot capture. For example, our models assume that people have perfect memory for the stimuli and exact recall of the values of the variables presented on each trial.These assumptions are clearly false, and a more realistic treatment of memory and perception might make it possible to tease apart the assumptions in our model that are due to these factors (e.g., high noise parameters) from those that capture human inductive inference (e.g., the set of kernels appearing in the mixture).

There are going to be aspects of human performance that cannot be captured by the kind of computational-level models we have considered, such as sensitivity to the order in which stimuli are presented, that may be candidates for identifying algorithmic-level implementations of these ideal solutions (similar to the role of order effects in categorization Anderson, 1991; Sanborn et al., 2010). As a starting place, it may be worth drawing inspiration from efforts to overcome the difficulty of scaling Gaussian processes to large data sets in the face of limited memory (e.g. Hensman, Fusi, & Lawrence, 2013).

If future work is to provide a deeper understanding of function learning, including the roles played by priors (and free parameters more generally), learners' limited cognitive resources, and individual differences, it will be necessary to go beyond the evaluation methods that have become standard in function learning, in at least two respects. First, it is now increasingly feasible and important to examine not just overall error rates, or aggregate correlations between model predictions and human judgments, but the accuracy with which a model can predict human judgments for individual points, given the values and order of previous training and test points. In addition to making it possible to assess how well a model can account for the process and dynamics of function learning – which include order effects as described above, as well as other phenomena, like the tacit belief that a relationship might be changing over time or trials (Speekenbrink & Shanks, 2010) – such an approach is more robust to aggregation artifacts (Navarro, Griffiths, Steyvers, & Lee, 2006). Second, we have taken a common approach to fitting and testing cognitive models – finding global parameters or priors that give low error or a high likelihood of the experimental data – but this approach has drawbacks beyond the simple risk of overfitting. Perhaps the most serious of these, in cases where we are interested in the priors that people tacitly use, is that this approach licenses only coarse-grained conclusions about what priors are likely or plausible given the experimental data. In the future, we hope that cheaper computational resources and increasingly efficient algorithms will make it feasible to conduct a Bayesian analysis of our model and others, which would provide a clearer picture of the priors that are consistent with group-level tendencies as well as individual differences (Hemmer, Tauber, & Steyvers, 2014).

Finally, a key question for any Bayesian model of cognition is the origins of the inductive biases that are expressed in the prior distribution. Having established a picture of adult inductive biases at the start of an experiment, we can begin to explore questions related to the development of these inductive biases. Within the Bayesian framework, it is possible to make inferences at the level of prior distributions by using hierarchical Bayesian models (Tenenbaum, Griffiths, & Kemp, 2006). In the case of our Gaussian process model, people could learn the set of kernels or parameter distributions for flexible kernel types (for work related to these ideas, see Wilson & Adams, 2013; Duvenaud, Lloyd, Grosse, Tenenbaum, & Ghahramani, 2013), the probabilities assigned to those kernels, and other parameters of the model. The predictions of this account of the origins of human inductive biases for function learning can be evaluated by comparing the performance of children and adults in function learning tasks and conducting transfer learning experiments examining how people's inductive biases change through experience, and is an exciting direction for future research.

Conclusion

We have presented a rational account of human function learning, drawing on ideas from machine learning and statistics to show that the two approaches that have dominated previous work – rules and similarity – can be interpreted as two views of the same kind of optimal solution to this problem. Our Gaussian process models combine the strengths of both approaches, using a mixture of kernels to allow systematic extrapolation as well as sensitive non-linear interpolation. Tests of the performance of this model on benchmark datasets show that it can capture some of the basic phenomena of human function learning, and is competitive with existing process models. The result is a clear characterization of human inductive biases for function learning, and a new set of links between human learning and ideas in statistics and machine learning.

References

Aldous, D. (1985). Exchangeability and related topics. In *École d'été de probabilités de Saint-Flour, XIII—1983* (pp. 1–198). Berlin: Springer.

Anderson, J. R. (1990). *The adaptive character of thought*. Hillsdale, NJ: Erlbaum.

Anderson, J. R. (1991). The adaptive nature of human categorization. *Psychological Review*, *98*(3), 409–429.

Ashby, F. G., & Alfonso-Reese, L. A. (1995). Categorization as probability density estimation. *Journal of Mathematical Psychology*, *39*, 216-233.

Bernardo, J. M., & Smith, A. F. M. (1994). *Bayesian theory*. New York: Wiley.

Borji, A., & Itti, L. (2013). Bayesian optimization explains human active search. In C. Burges, L. Bottou, M. Welling, Z. Ghahramani, & K. Weinberger (Eds.), *Advances in neural information processing systems 26* (pp. 55–63). Curran Associates, Inc. Retrieved from http://papers.nips.cc/paper/4952-bayesian-optimization-explains-human-a

Bott, L., & Heit, E. (2004). Nonmonotonic extrapolation in function learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *30*(1).

Brehmer, B. (1971). Subjects' ability to use functional rules. *Psychonomic Science*, *24*, 259-260.

Brehmer, B. (1974). Hypotheses about relations between scaled variables in the learning of probabilistic inference tasks. *Organizational Behavior and Human Decision Processes*, *11*, 1-27.

Brehmer, B. (1976). Subjects' ability to find the parameters of functional rules in probabilistic inference tasks. *Organizational Behavior and Human Performance*, *17*(2), 388–397.

Brehmer, B., Alm, H., & Warg, L. (1985). Learning and hypothesis testing in probabilistic inference tasks. *Scandinavian journal of psychology*, *26*(1), 305–313.

Busemeyer, J. R., Byun, E., DeLosh, E. L., & McDaniel, M. A. (1997). Learning functional relations based on experience with input-output pairs by humans and artificial neural networks. In K. Lamberts & D. Shanks (Eds.), *Concepts and categories* (p. 405-437).

Cambridge: MIT Press.

Byun, E. (1995). Interaction between prior knowledge and type of nonlinear relationship on function learning. *Unpublished doctoral dissertation, Purdue University*.

Carroll, J. (1963). *Functional learning: The learning of continuous functional mappings relating stimulus and response continua*. Princeton, NJ: Educational Testing Service.

Chater, N., & Vitanyi, P. (2003). Simplicity: a unifying principle in cognitive science. *Trends in Cognitive Science*, *7*, 19-22.

DeLosh, E. L., Busemeyer, J. R., & McDaniel, M. A. (1997). Extrapolation: The sine qua non of abstraction in function learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *23*, 968-986.

Duvenaud, D., Lloyd, J. R., Grosse, R., Tenenbaum, J. B., & Ghahramani, Z. (2013). Structure discovery in nonparametric regression through compositional kernel search. *arXiv preprint arXiv:1302.4922*.

Erickson, M., & Kruschke, J. (1998). Rules and exemplars in category learning. *Journal of Experimental Psychology: General*, *127*(2), 107.

Gelman, A., Carlin, J., Stern, H., & Rubin, D. (2004). *Bayesian data analysis*. CRC press.

Gilks, W., Richardson, S., & Spiegelhalter, D. J. (Eds.). (1996). *Markov chain Monte Carlo in practice*. Suffolk, UK: Chapman and Hall.

Griffiths, T. L., Lucas, C. G., Williams, J. J., & Kalish, M. L. (2009). Modeling human function learning with Gaussian processes. *Advances in Neural Information Processing Systems*, *21*.

Griffiths, T. L., Vul, E., & Sanborn, A. (2012). Bridging levels of analysis for probabilistic models of cognition. *Current Directions in Psychological Science*, *21*(4), 263–268.

Hemmer, P., Tauber, S., & Steyvers, M. (2014). Moving beyond qualitative evaluations of bayesian models of cognition. *Psychonomic bulletin & review*, 1–15.

Hensman, J., Fusi, N., & Lawrence, N. D. (2013). Gaussian processes for big data. *arXiv preprint arXiv:1309.6835*.

Jacobs, R., Jordan, M., Nowlan, S., & Hinton, G. (1991). Adaptive mixtures of local experts.

Neural computation, *3*(1), 79–87.

Kalish, M. L. (2013). Learning and extrapolating a periodic function. *Memory & cognition*, *41*(6), 886–896.

Kalish, M. L., Griffiths, T. L., & Lewandowsky, S. (2007). Iterated learning: Intergenerational knowledge transmission reveals inductive biases. *Psychonomic Bulletin and Review*.

Kalish, M. L., Lewandowsky, S., & Kruschke, J. (2004). Population of linear experts: Knowledge partitioning and function learning. *Psychological Review*, *111*, 1072-1099.

Kirby, S. (2001). Spontaneous evolution of linguistic structure: An iterated learning model of the emergence of regularity and irregularity. *IEEE Journal of Evolutionary Computation*, *5*, 102-110.

Koh, K., & Meyer, D. (1991). Function learning: Induction of continuous stimulus-response relations. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *17*(5), 811–836.

Kwantes, P., & Neal, A. (2006). Why people underestimate y when extrapolating in linear functions. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *32*(5), 1019.

Lewandowsky, S. L., Kalish, M. L., & Ngang, S. K. (2002). Simplified learning in complex situations: Knowledge partitioning in human learning. *Journal of Experimental Psychology: General*, *131*, 163-193.

Lindsey, R. V., Mozer, M. C., Huggins, W. J., & Pashler, H. (2013). Optimizing instructional policies. In C. Burges, L. Bottou, M. Welling, Z. Ghahramani, & K. Weinberger (Eds.), *Advances in neural information processing systems 26* (pp. 2778–2786). Curran Associates, Inc. Retrieved from `http://papers.nips.cc/paper/4887-optimizing-instructional-policies.pdf`

Little, D. R., & Shiffrin, R. M. (2009). Simplicity bias in the estimation of causal functions. In *Proceedings of the thirty-first annual conference of the cognitive science society* (pp. 1157–1162).

Lucas, C. G., Sterling, D. J., & Kemp, C. (2012). Superspace extrapolation reveals inductive biases in function learning. In N. Miyake, D. Peebles, & R. P. Cooper (Eds.), *Proceedings of the 34th annual conference of the cognitive science society.* Cognitive Science Society.

MacKay, D. (1995). Probable networks and plausible predictions - a review of practical bayesian methods for supervised neural networks. *Network: Computation in Neural Systems*, *6*, 469-505.

Marr, D. (1982). *Vision*. San Francisco, CA: W. H. Freeman.

McDaniel, M., & Busemeyer, J. (2005). The conceptual basis of function learning and extrapolation: Comparison of rule-based and associative-based models. *Psychonomic bulletin & review*, *12*(1), 24.

McDaniel, M., Dimperio, E., Griego, J., & Busemeyer, J. (2009). Predicting transfer performance: A comparison of competing function learning models. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *35*(1), 173.

Meeds, E., & Osindero, S. (2006). An alternative infinite mixture of gaussian process experts. In Y. Weiss, B. Schölkopf, & J. Platt (Eds.), *Advances in neural information processing systems 18* (pp. 883–890). Cambridge, MA: MIT Press.

Minh, H., Niyogi, P., & Yao, Y. (2006). MercerâĂŹs theorem, feature maps, and smoothing. *Learning theory*, 154–168.

Mitchell, T. M. (1997). *Machine learning*. New York: McGraw Hill.

Navarro, D. J., Griffiths, T. L., Steyvers, M., & Lee, M. D. (2006). Modeling individual differences using dirichlet processes. *Journal of Mathematical Psychology*, *50*, 101-122.

Neal, R. M. (1994). *Priors for infinite networks* (Tech. Rep.). Technical Report CRG-TR-94-1, Department of Computer Science, University of Toronto, 1994.

Neal, R. M. (1998). *Markov chain sampling methods for Dirichlet process mixture models* (Tech. Rep. No. 9815). Department of Statistics, University of Toronto.

Rasmussen, C. E., & Ghahramani, Z. (2002). Infinite mixtures of gaussian process experts. In T. Dietterich, S. Becker, & Z. Ghahramani (Eds.), *Advances in neural information processing*

systems 14 (pp. 881–888). MIT Press.

Rasmussen, C. E., & Williams, C. K. I. (2006). *Gaussian processes for machine learning*. MIT Press.

Sanborn, A. N., Griffiths, T. L., & Navarro, D. J. (2010). Rational approximations to rational models: alternative algorithms for category learning. *Psychological Review*, *117*(4), 1144.

Shepard, R. N. (1987). Towards a universal law of generalization for psychological science. *Science*, *237*, 1317-1323.

Speekenbrink, M., & Shanks, D. R. (2010). Learning in a changing environment. *Journal of Experimental Psychology: General*, *139*(2), 266.

Tenenbaum, J. B., Griffiths, T. L., & Kemp, C. (2006). Theory-based Bayesian models of inductive learning and reasoning. *Trends in Cognitive Science*, *10*, 309-318.

Williams, C. K. I. (1998). Prediction with gaussian processes: From linear regression to linear prediction and beyond. In M. I. Jordan (Ed.), *Learning in graphical models* (p. 599-621). Cambridge, MA: MIT Press.

Wilson, A. G., & Adams, R. P. (2013). Gaussian process covariance kernels for pattern discovery and extrapolation. *arXiv preprint arXiv:1302.4245*.

Appendix A

Equivalence of Bayesian linear regression and Gaussian processes

This appendix contains a more detailed description of how Bayesian linear regression can be expressed using Gaussian processes, showing first that Bayesian linear regression with normal priors on coefficients can be represented as a mean function and a covariance function relating observed and predicted y values. Next we show that such a representation – which describes a Gaussian process – can be used for prediction directly, setting aside the linear regression interpretation.

Bayesian linear regression

In Bayesian linear regression, the goal is to use n observed x-values, $\mathbf{x}_n = (x_1, ..., x_n)$, and their corresponding y-values with added noise, $\mathbf{t}_n = (t_1, ..., t_n)$, to predict y_{n+1} from x_{n+1}. Let the hypothesis space include linear functions of the form $y = b_0 + b_1 x$, where the prior probability of a given function is a multivariate Gaussian distribution on $\mathbf{b} = (b_0, b_1)$ with mean zero and covariance Σ_b. Applying Equation 1 then results in a multivariate Gaussian posterior distribution on $\mathbf{b}$ (see Bernardo & Smith, 1994) with

$$E[\mathbf{b}|\mathbf{x}_n, \mathbf{t}_n] = (\sigma_t^2 \Sigma_b^{-1} + \mathbf{X}_n^T \mathbf{X}_n)^{-1} \mathbf{X}_n^T \mathbf{t}_n \tag{9}$$

$$\text{cov}[\mathbf{b}|\mathbf{x}_n] = (\Sigma_b^{-1} + \frac{1}{\sigma_t^2} \mathbf{X}_n^T \mathbf{X}_n)^{-1} \tag{10}$$

where $\mathbf{X}_n = [\mathbf{1}_n \mathbf{x}_n]$ (i.e., a matrix with a vector of ones horizontally concatenated with $\mathbf{x}_{n+1}$) Since $\mathbf{y}_{n+1}$ is simply a linear function of $\mathbf{b}$, applying Equation 2 yields a Gaussian predictive distribution, with y_{n+1} having mean $[1\ x_{n+1}] E[\mathbf{b}|\mathbf{x}_n, \mathbf{t}_n]$ and variance $[1\ x_{n+1}]\text{cov}[\mathbf{b}|\mathbf{x}_n][1\ x_{n+1}]^T$. The predictive distribution for t_{n+1} is similar, but with the addition of σ_t^2 to the variance.

In the more general case where y is a function of an arbitrary number of basis functions $\phi^{(1)}, ..., \phi^{(k)}$ of $\mathbf{x}$, the same result holds, substituting $\mathbf{\Phi} = [\mathbf{1}_n\ \phi^{(1)}(\mathbf{x}_n)\ ...\ \phi^{(k)}(\mathbf{x}_n)]$ for $\mathbf{X}$ and $[1\ \phi^{(1)}(x_{n+1})\ ...\ \phi^{(k)}(x_{n+1})]$ for $[1\ x_{n+1}]$, where $\phi(\mathbf{x}_n) = [\phi(x_1)\ ...\ \phi(x_n)]^T$.

Gaussian processes

The Gaussian process approach amounts to predicting y using $\mathbf{x}$ by defining a joint Gaussian distribution on $\mathbf{y}_{n+1}$ given $\mathbf{x}_{n+1}$ and conditioning on $\mathbf{y}_n$, with covariance matrix

$$\mathbf{K}_{n+1} = \begin{pmatrix} \mathbf{K}_n & \mathbf{k}_{n,n+1} \\ \mathbf{k}_{n,n+1}^T & k_{n+1} \end{pmatrix} \tag{11}$$

where $\mathbf{K}_n$ depends on the values of $\mathbf{x}_n$, $\mathbf{k}_{n,n+1}$ depends on $\mathbf{x}_n$ and x_{n+1}, and k_{n+1} depends only on x_{n+1}. If we condition on $\mathbf{y}_n$, the distribution of y_{n+1} is Gaussian with mean $\mathbf{k}_{n,n+1}^T \mathbf{K}_n^{-1} \mathbf{y}$ and variance $k_{n+1} - \mathbf{k}_{n,n+1}^T \mathbf{K}_n^{-1} \mathbf{k}_{n,n+1}$. This approach to prediction uses a *Gaussian process*, a stochastic process that induces a Gaussian distribution on $\mathbf{y}$ based on the values of $\mathbf{x}$. We can extend this approach to predict y_{n+1} from x_{n+1}, $\mathbf{t}_n$, and $\mathbf{x}_n$ by adding $\sigma_t^2 \mathbf{I}_n$ to $\mathbf{K}_n$, where $\mathbf{I}_n$ is the $n \times n$ identity matrix, to take into account the additional variance associated with the observations $\mathbf{t}_n$.

The covariance matrix $\mathbf{K}_{n+1}$ is specified using a two-place function in x known as a *kernel*, with $K_{ij} = K(x_i, x_j)$. Any kernel that results in an appropriate (symmetric, positive-definite) covariance matrix for all $\mathbf{x}$ can be used. Common kinds of kernels include radial basis functions, e.g.,

$$K(x_i, x_j) = \theta_1^2 \exp(-\frac{1}{\theta_2^2}(x_i - x_j)^2) \tag{12}$$

with values of y for which values of x are close being correlated, and periodic functions, e.g.,

$$K(x_i, x_j) = \theta_3^2 \exp(\theta_4^2(\cos(\frac{2\pi}{\theta_5}[x_i - x_j]))) \tag{13}$$

indicating that values of y for which values of x are close relative to the period θ_3 are likely to be highly correlated. Gaussian processes thus provide a flexible approach to prediction, with the kernel defining which values of x are likely to have similar values of y.

Appendix B

Priors and parameters

The Gaussian process model makes use of two kinds of prior distributions: priors over different types kernels, and priors over the parameters of the individual kernel functions. The prior over kernels reflects past results showing that people act in a manner consistent with the assumption that positive linear relationships as more likely than negative linear relationships, which are more likely than quadratic relationships, which are in turn more likely than arbitrary non-linear relationships (Brehmer, 1974).

In previous experiments designed to reveal prior beliefs about the prevalence of different kinds of relationships (Kalish et al., 2007), positive linear relationships are approximately 8 times as likely as negative linear relationships, but fewer specifics are available for the rates at which people generate other relationships, beyond the qualitative ordering described by Busemeyer et al (Busemeyer et al., 1997). As a result, we choose prior probabilities proportional to 8,1, 0.1, and 0.01 for positive linear, negative linear, quadratic, and radial basis kernels.

The parameters for the kernels were given gamma distributed priors, and included the variances of the weights and intercept for the linear and quadratic kernels, the height and distance parameters for the radial basis kernel. In all of these cases, the gamma distribution had a shape parameter of 1.001, which had the effect of discounting values very close to zero. All of the scale parameters were set to one, except for the radial basis function's width, or smoothness.

As discussed in the body text, Models 2 and 3, which are mixtures of Gaussian processes and mixtures of Gaussian process experts, respectively, also included a parameter α determining how dispersed points were expected to be over distinct experts. For Model 3, the prior over x for each expert was specified by assuming two virtual points at the extremes of the x range, and had no free parameters.

Model fitting

For the fitted parameters, we considered combinations of $\sigma_t^2 \in \{0.001, 0.01, 0.05, 0.1, 0.2\}$, $\alpha \in \{0.01, 0.1, 1, 10\}$, and $\theta_l \in \{1, 10\}$, where σ_t^2 is the variance of points around their true function, α is the dispersion parameter for the Chinese Restaurant Process prior on partitions, and θ_l controls the smoothness of functions under the radial basis kernel. In all cases, we used parameters that maximized the mean correlation between model predictions and mean human judgments across the difficulty-of-learning and extrapolation data. The remaining parameters, for variances for non-noise terms and the radial basis function's height scale, were all fixed at 1. Separate fits were obtained for the POLE data. See Table B1 for the values that were applied to the interpolation and extrapolation experiments, and Table B2 for the values that were applied to the knowledge partitioning and iterated learning experiments.

Table B1
Model Parameters for Interpolation and Extrapolation Phenomena.

	σ_t^2	θ_l	α
Model 1, Linear	0.01	NA	NA
Model 1, Quadratic	0.01	NA	NA
Model 1, Radial-basis	0.10	1	NA
Model 1, LQ	0.01	NA	NA
Model 1, LR	0.10	1	NA
Model 1, RQ	0.10	1	NA
Model 1, LRQ	0.05	10	NA
Model 2	0.01	10	1.00
Model 3	0.01	10	1.00

Note: Model 1 kernels included combinations of linear (L), quadratic (Q), and radial basis functions (R).

Table B2
Model Parameters for Iterated Learning and Knowledge Partitioning Phenomena.

	σ_t^2	θ_l	α
Model 1	0.2	10	NA
Model 2	0.001	10	10
Model 3	0.001	10	10

Note: Only models incorporating all kernels were considered.

Appendix C

Inference

This appendix describes the procedures by which we obtained predictions for each of our models.

Gaussian process model (Model 1)

To obtain predictions, we performed probabilistic inference using a Markov chain Monte Carlo (MCMC) algorithm for an introduction, see Gilks, Richardson, and Spiegelhalter (1996). This algorithm defines a Markov chain for which the stationary distribution is the distribution from which we wish to sample. In our case, this is the posterior distribution over types and the hyperparameters for the kernels $\boldsymbol{\theta}$ given the observations $\mathbf{x}$ and $\mathbf{t}$. The hyperparameters include all kernel parameters discussed above and the noise in the observations σ_t^2. Our MCMC algorithm repeats two steps. The first step is sampling the type of function conditioned on $\mathbf{x}$, $\mathbf{t}$, and the current value of θ, with the probability of each type being proportional to the product of $p(\mathbf{t}_n|\mathbf{x}_n)$ for the corresponding Gaussian process and the prior probability of that type as given by π. The second step is sampling the value of θ given $\mathbf{x}_n$, $\mathbf{t}_n$, and the current type, which is done using a Metropolis-Hastings procedure, proposing a value for θ from a Gaussian distribution centered on the current value and deciding whether to accept that value based on the product of the probability it assigns to $\mathbf{t}_n$ given $\mathbf{x}_n$ and the prior $p(\theta)$. In all cases, this inference procedure was iterated 8000 times.

Mixtures of Gaussian process experts (Models 2 and 3)

The infinite mixture of Gaussian process model extends the basic model by assigning observations to different experts, or Gaussian processes. The prior probability that an observation will be assigned to a particular expert is determined by a Chinese restaurant process (CRP) prior, where the probability that a new point will be assigned to an expert is proportional to the number of points already assigned to it, and the probability that a point will be assigned to a new expert is determined by α: if expert k has n_k points of a total N assigned points, a new point will be assigned to it with probability $\frac{n_k}{N+\alpha}$, and to a new table with probability $\frac{\alpha}{N+\alpha}$. For Model 3, the

locations of points in x also influence the experts to which they are assigned: each expert is assumed to have a Gaussian distribution over x values with a minimally informative prior, defined by combining improper constant priors for the mean and variance with two virtual points at 0 and 1, the extremes of the range of x. This prior leads to a t-distributed density for new points, conditional on those already assigned to the expert (for details, see Gelman, Carlin, Stern, & Rubin, 2004).

The first steps of the inference procedure are identical to those used for Model 1. These are followed by Gibbs sampling for the assignments of points to experts, resampling each point and assigning it to a new or existing expert according to its conditional probability under that expert given all other points and all parameters (Neal, 1998). Simulated annealing was used to speed mixing of the sampling chains, which included 8000 iterations of each step.